# Table of Contents

# Storage Best Practices

## Portable File Names and Sizes

## DRAFT

This article is being reviewed for completeness and technical accuracy.

### Portable File Names

Use portable file names. A name is portable if it contains only ASCII letters and digits, `.', `_', and `-'. Do not use spaces or wildcard characters or start with a `-' or `//', or contain `/-'. Avoid deep directory nesting.

If you intend to have tar archives to be read under MSDOS, you should not rely on case distinction for file names, and you might use the GNU **doschk** program for helping you further diagnose illegal MSDOS names, which are even more limited than Unix like operating system.

### Portable File Sizes

Even though Lou's archive filesystem will allow a file size to be greater than several hundred gigabytes, not all operating systems or filesystems can manage this much data in a single file. If you plan to transfer files to an old Mac or PC desktop you may want to verify the maximum filesize it will support. Likely a single file will need to be less than 4 GB before it will transfer successfully.

# Dealing with Slow File Retrieval

## DRAFT

This article is being reviewed for completeness and technical accuracy.

There are sometimes problems with commands on Lou, that should finish quickly, but end up taking a long time.

When you do an " **ls** " on Lou, you see all the files on disk that you've put there. However, most of the files are actually written to tape using SGI's Data Migration Facility (DMF).

One problem with DMF is that it does not deal well with retrieving one file at a time from a long list of files. If you do an " **scp** " with a list of files, Unix feeds those files to DMF one at a time. This means that the tape(s) containing the files is getting constantly loaded and unloaded which is bad for the tape and tape drive, and also very slow. As the list of files gets longer (by use of "*" or moving a "tree" of files) the problem grows to where it can take hours to transfer a set of files that would only take a few minutes if they were on disk. When several people do file transfers at once that retrieve files one at a time, it can tie the system in knots.

### Optimizing File Retrieval

DMF let you fetch files to disk as a group with the **dmget** command. The tape is read once and gets all the requested files in a single pass. Essentially, give **dmget** the same list of files you are about to transfer, and when the **dmget** completes, then **scp**/**ftp**/**cp** the files as you had originally intended. Or you can put the **dmget** in the background and run your transfer while **dmget** is working. If any files are already on disk, dmget sees this and doesn't try to get them from tape.

There is also a **dmfind** command that let you walk a file tree to find offline files to give to **dmget** . Make very sure you are in the correct directory before running **dmfind** . Use the " pwd " command to determine your current directory.

Please check to make sure too much data isn't brought back online at once by using du with the *--apparent-size* option or by using **/usr/local/bin/dmfdu** .

Note that **dmfdu** will give an error message for each symbolic link that points at a nonexistent file.

```
lou% /usr/local/bin/dmfdu Foo
    Foo
             13 MB regular          340 files
           1114 MB dual-state      1920 files
          74633 MB offline         2833 files
             13 MB small            340 files
          75761 MB total            5093 files
```

When transferring data between Lou and Columbia nodes use the /nobackup filesystems, instead of the Columbia NFS (slow) home directories.

File transfer rates vary depending on the load on the system and how many users are transferring files at the same time. Transferring files using **scp** between Lou and Columbia nodes on the /nobackup file system for files larger than 100 megabytes is typically between 7 - 17 MB/s using the gigabit network interface. Transferring files using scp between the Columbia nodes for files larger than 100 megabytes, is typically between 20-30 MB/s for the gigabit network interface.

### Example 1:

```
lou%  dmget *.data &
lou%  scp -qp *.data  myhost.jpl.nasa.gov:/home/user/wherever
```

### Example 2:

```
lou%  dmfind  /u/username/FY2000 -state OFL -print | dmget &
lou%  scp -rqp /u/username/FY2000 some_host:/nobackup/user1/whereever
```

You can see the state of a file by doing " **dmls -l**" instead of " **ls -l**". For more information on using DMF, please look at:

Data-Migration-Facility-DMF-Commands_250.html

### Maximum Amount of Data to Retrieve Online

The online disk space for Lou1-3 is much, much less than its tape storage capacity, and it is impossible to retrieve all files to online storage at the same time. So, before retrieving a large amount of data, you should check that there is enough online space for it. The **df** command shows the amount of free space in a filesystem. The Lou script **dmfdu** reports how much total (online and offline) data is in a directory. To use this script, simply " **cd** " into the directory you want to know total amount of data for all the files in the current directory and execute the script.

If you would like to know the total amount of data under your home directory on Lou, you need to first find out if your account is under s1a-s1e, s2a-s2e or s3a-s3e. Assuming you are under s1b, you can then use **dmfdu /s1b/your_userid** to find the total amount. Another alternative is to simply cd to your home directory and use " **dmfdu *** ", which will show use for each file or directory.

Lou1-3's archive filesystems are between 8 TB and 30 TB in size, but the available space typically floats between 10% to 30%. In the example 3, 29% of space is unused. It is best to retrieve at most 10% of the filesystem space at a time. Do what you need to with those files (scp, edit, compile, etc), then release ( dmput -r ) the space, and then retrieve the next group of files, use them, then release the space, etc. For example 3, retrieve one directory's data from tape, copy the data to remote host then release the data blocks, before retrieving more data from tape.

**Example 3:**

```
lou% df -lh .
Filesystem            Size  Used Avail Use% Mounted on
/dev/lxvm/lsi_s1b     8.6T  6.1T  2.6T  71% /s1b



lou% dmfdu project1 project2
project1
          2 MB regular            214 files
         13 MB dual-state           1 files
     229603 MB offline            101 files
          2 MB small              214 files
     229606 MB total              315 files

project2
          7 MB regular            245 files
       4661 MB dual-state          32 files
     218999 MB offline             59 files
          7 MB small              245 files
     223668 MB total              336 files


lou% cd project1

lou% dmfind . -state OFL -print | dmget &

lou% scp -rp /u/username/project1 remote_host:/nobackup/username

(Verify that the data has successfully transferred)

lou% dmfind . -state DUL -print | dmput -rw

lou% df -lh .


lou% cd ../project2

lou% dmfind . -state OFL -print | dmget &

lou% scp -rpq /u/username/project2 remote_host:/nobackp/username

lou% dmfind . -state DUL -print  | dmput -rw
```